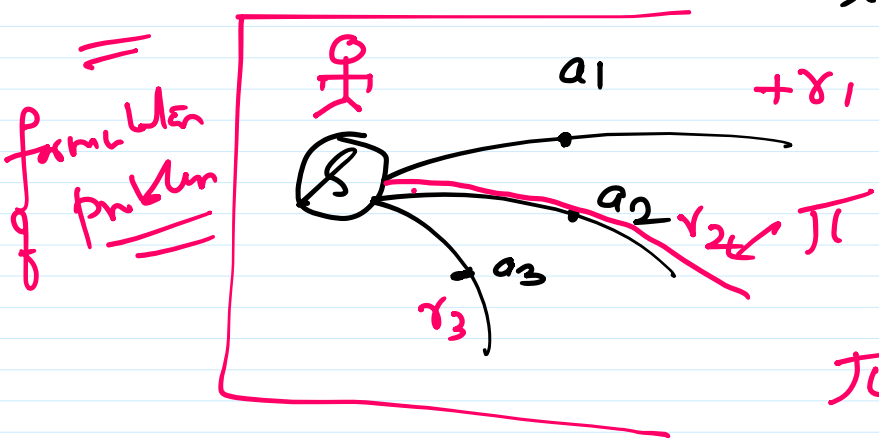
02 August 2026 10:37

Agenda

Multi-arm Bandit problem

Goal of RL is to come up with

$\pi(a|s)$

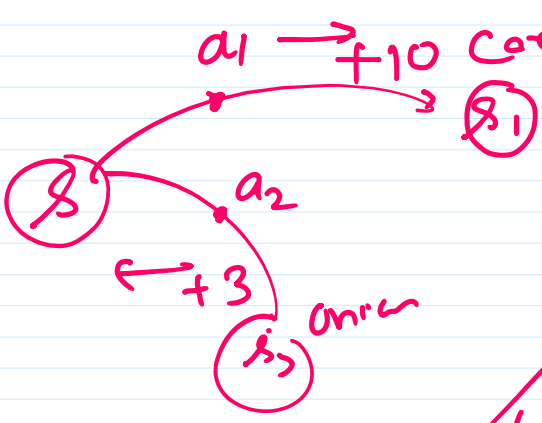


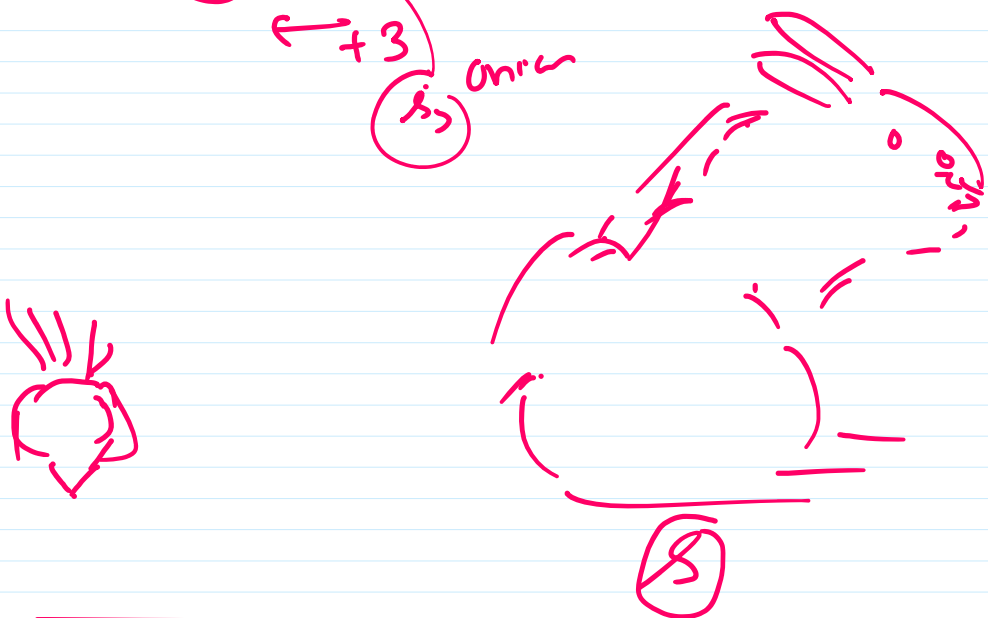
$\pi(a_1|s)$

$\pi(a_2|s)$

$\pi(a_3|s)$

π_θ → Model parameters





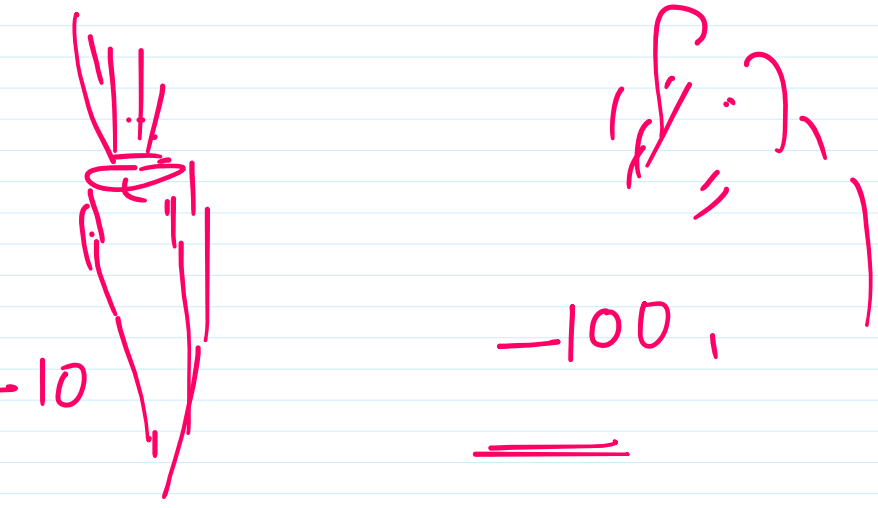
Policy

= 0.3

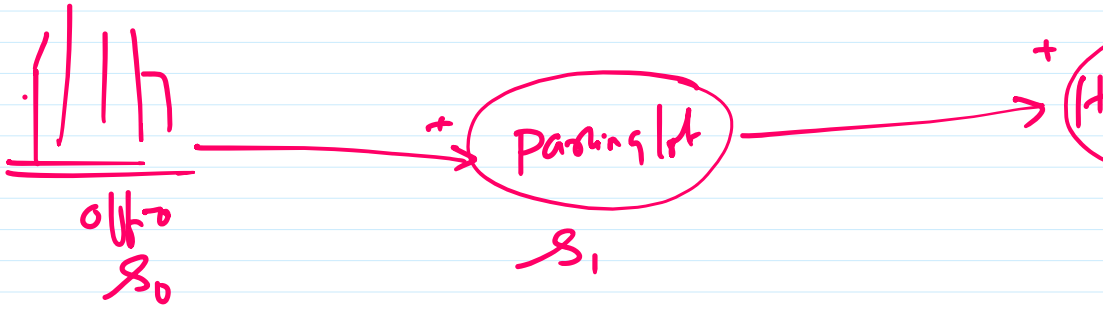
: 0.4

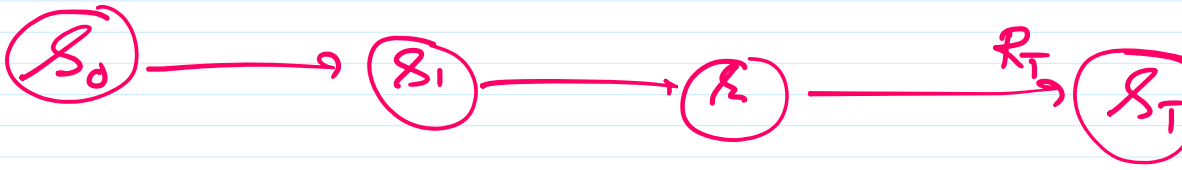
= 0.3

: Gradient descent / Similar algorithms



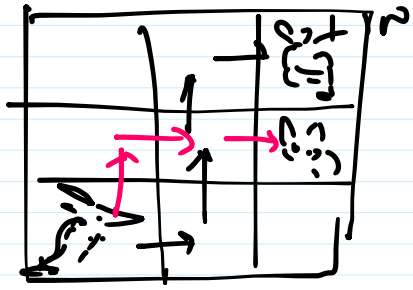
Ex2

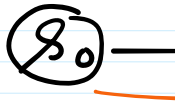




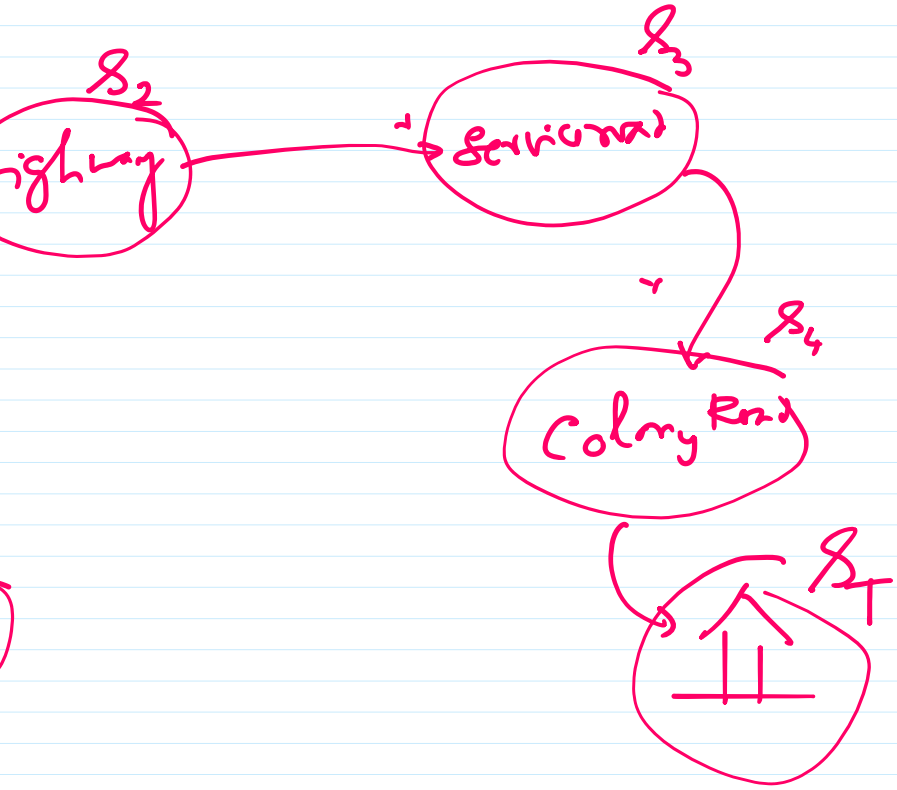
Episode

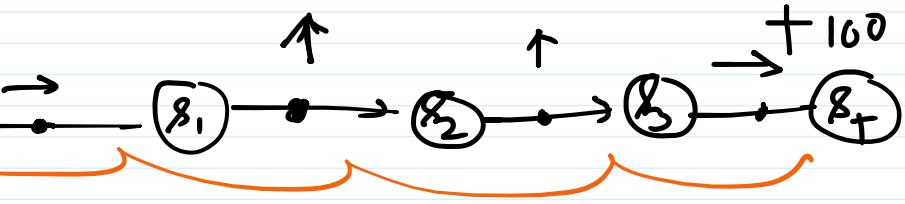
Grid world



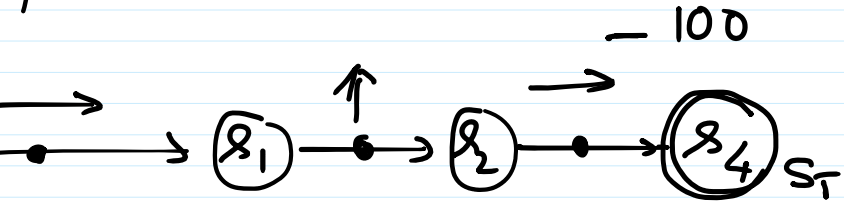


how can rat can make



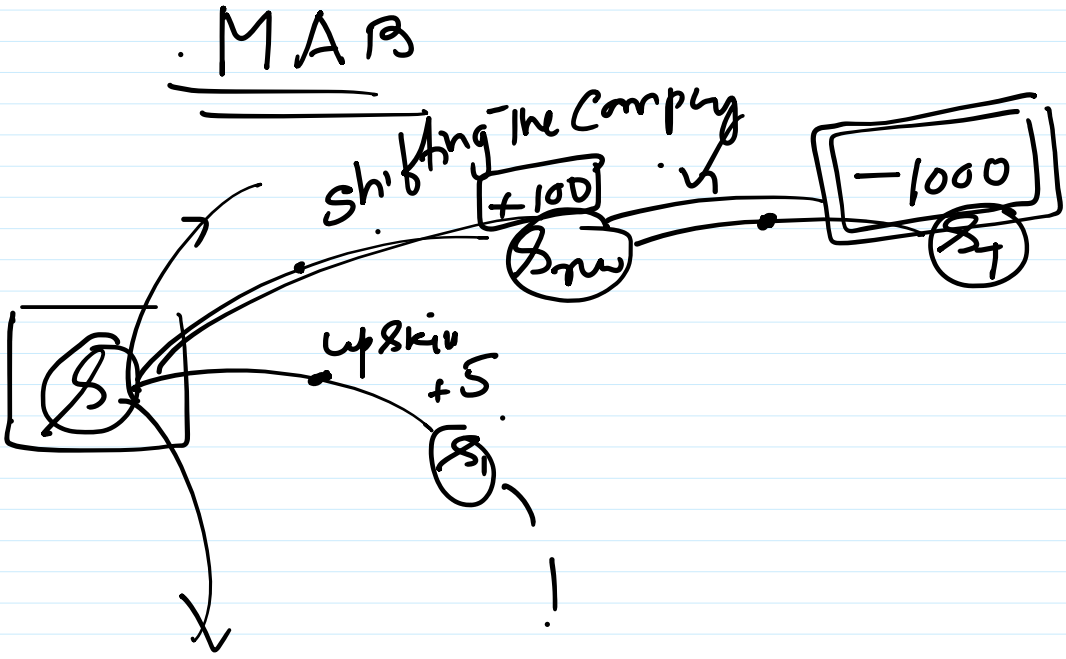


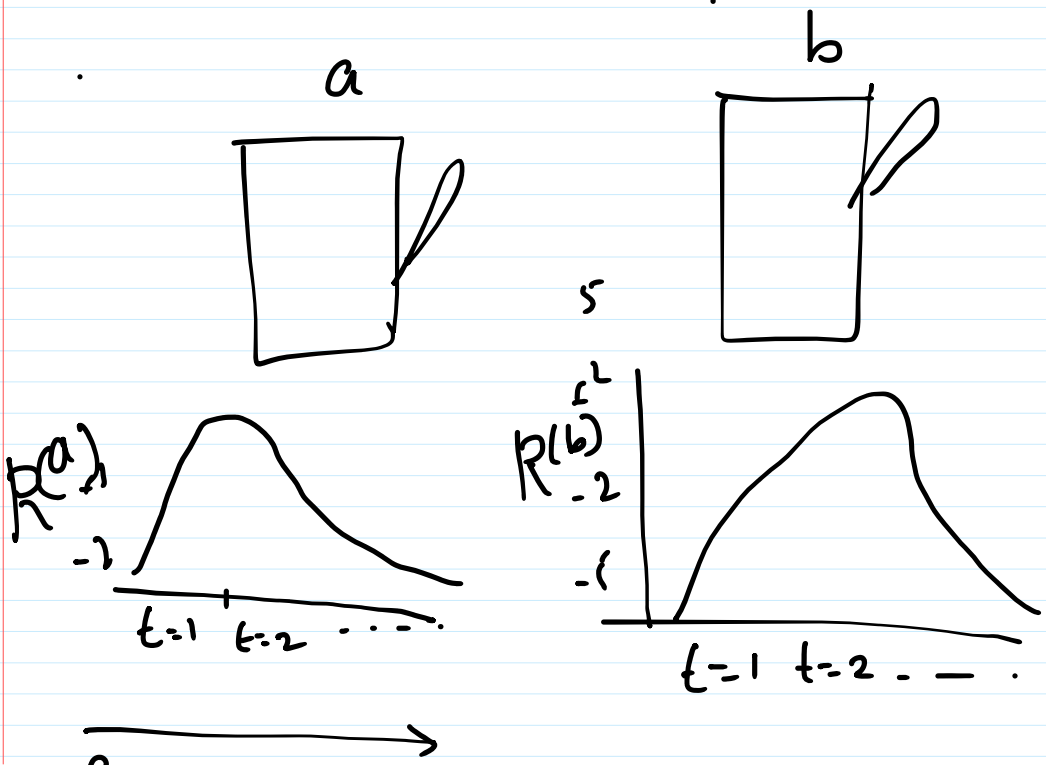
Episode 1



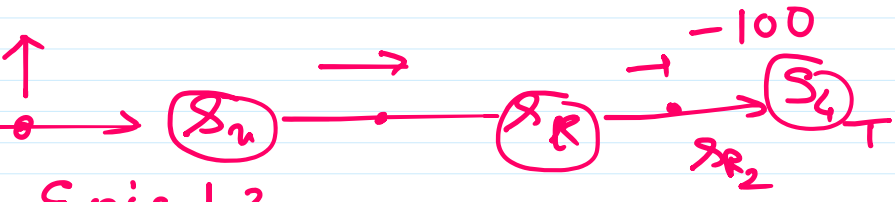
Episode 2

How can agent can make better decisions that yields "better" cumulative return of Reward



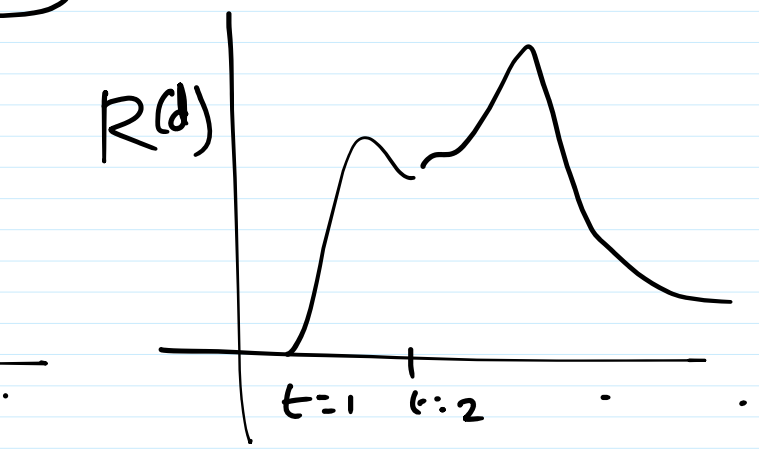


Episode 2



Episode 3

Exploration
Exploitation



Time steps

$$E(a) = \sum_{i=1}^{N} p(A_i = a) * R(a_i)$$

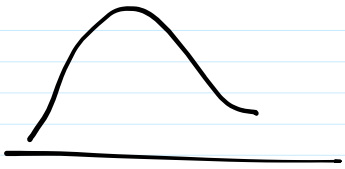
$$p(A = a)\, R(a)$$

$\frac{3}{10} \times R$

Consider N steps

At any step $p_r(A = a) = \frac{1}{N}$

In Rat problem

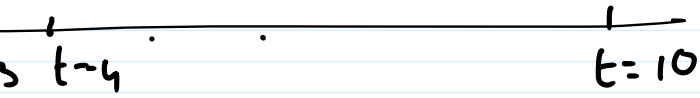


ε- greedy

Let ϵ : 0.2 → 100 time steps;

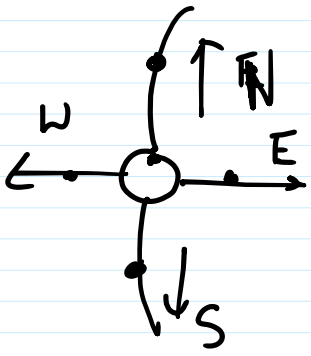
k = 2

$t = a$)



10 Selections

* $R(a) = \mathbb{E}[R \mid A = a]$



$\pi(a|s) = \frac{1}{4}$

20 time steps we are

exploring

time steps

$k = 2$

$\epsilon = 0.6$ → does

$\epsilon = 0.1$ ↗

Incremental implementation of action values

Consider avg method

$$Q_{n+1}(a) = \frac{1}{n} \sum_{i=1}^{n} R_i$$

R_i → Reward

Q_{n+1} → Estimated action value

n times

$$Q_{n+1}(a) = \frac{1}{n} \left[R_n + \sum_{i=1}^{n-1} R_i \right] \longrightarrow$$

R_n → Reward Received at

$$Q_{n+1}(a) = \frac{1}{n} \sum_{i=1}^{n} R_i$$

Replace n with $(n-1)$

$$Q_n = \frac{1}{(n-1)} \sum_{i=1}^{n-1} R_i$$

$$\Rightarrow \boxed{(n-1)\, Q_n = \sum_{i}}$$

Substitute (3) in (2)

10 time steps

il make Sense

$$\frac{1}{n} \times R_1 + \frac{1}{n} R_2 + \cdots \frac{1}{n} R_n$$

→ (1)

Reward Recieved after i^{Th} Selection of this action

alue after the action has been selected

mes

(2)

n^{Th} Selection of this action

$$\frac{\sum_{i=1}^{n-1} R_i}{n-1} \rightarrow (3)$$

Substitute (3) in (2)

$$Q_{n+1}(a) = \frac{1}{n}\left[R_n + (n-1)\,Q_n\right]$$

$$Q_{n+1}(a) = \frac{1}{n}\left[R_n + nQ_n - Q_n\right]$$

$$= \frac{R_n}{n} + Q_n - \frac{Q_n}{n}$$

$$\boxed{Q_{n+1} = Q_n + \frac{1}{n}\left(R_n - Q_n\right)} \rightarrow \text{Thi}$$

$$\boxed{Q_{n+1} = Q_n + \alpha\left(R_n - Q_n\right)} \qquad \boxed{\alpha =}$$

↳ step size

Non-stationary estimate of action value fn

$\Rightarrow$ giving maximum weightage to re

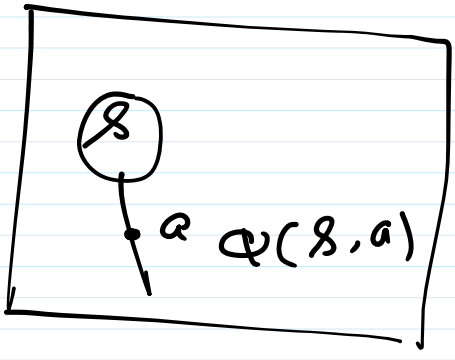
long past Rewards

$\Rightarrow$ penalize old rewards relative to new

s requires memory only for
$Q_{n,n}$

$\frac{1}{n}$

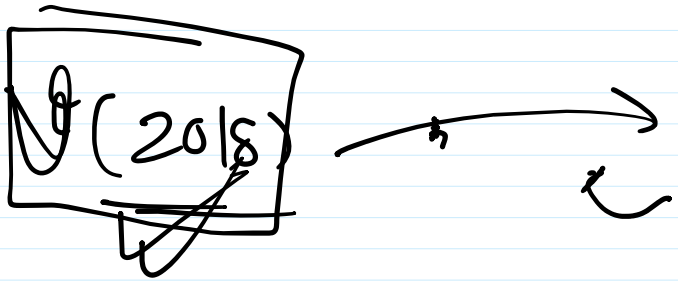
$Q(S,a)$



cent rewards than to

reward

⇒ penalize old rewards relative to new



reward